Appendix B

3D Dome: A 3D Digitization Testbed

Real experimentation into 3D digitization of dynamic events requires a system to record multiple video streams. These video streams must be synchronized, and the cameras must surround the scene to be modeled. This chapter explores the design and development of *3D Dome*, our facility for 3D digitization. This facility is designed to allow unknown, free-form, human-scale objects to move around freely in the work space. The design scales linearly with the number of cameras and uses only standard video equipment to minimize cost. The actual system was implemented with 51 video cameras on a 5-meter diameter geodesic dome.

B.1 Specifications

Any system for 3D digitization of dynamic, real-world events must satisfy three important requirements. First, the system must not interfere with the normal appearance or activity of the event, since the images will be used to construct both photometric and geometric scene models. This requirement prohibits the use of active lighting, which can improve

geometric modeling quality at the expense of distorting scene appearance. Second, the system must achieve sustained real-time performance in order to capture meaningful events. The precise definitions of *sustained* and *real-time* are context dependent. Third, the video must be captured digitally so that the 3D extraction processes can be applied.

In addition to these general system requirements, we added two objectives. First, the system should allow unknown, free-form, human-scale objects to move around freely in the work space. Human motion can occur relatively quickly, so we define *real-time* operation to mean full NTSC frame rate video capture of multiple video streams. We define *sustained* performance to mean video capture with no missed frames for at least one minute, enough to allow several repetitions of simple motions without restarting the system. Second, the system should be scalable so that nearly any number of cameras can be used.

B.2 Multi-Camera Video Capture Architecture

One obvious approach to solving this design problem is to directly capture digital video to some high-bandwidth, high-capacity storage device(s). However, synchronous digital acquisition of multiple video streams is a difficult task due to the magnitude of data involved. Just a single color camera generates 27 MBytes of raw data for 30 frames of size 480x640 per second. A 50-camera system recording for one minute would generate about 80 GBytes of data. Although specialized systems can provide the necessary bandwidth for a few video streams, extending these systems to many cameras would be prohibitively expensive. (Note that although lossy image compression would reduce the amount of data, the information loss may degrade the performance of the subsequent modeling processes. Lossless compression may help reduce the bandwidth, but usually only by a factor of 2-3, and at the cost of relatively large computation.)

Our alternative to direct digital acquisition is to use real-time analog acquisition and off-line digitization. Although the final capacity requirements remain unchanged, the digital acquisition can now occur at whatever rate is available. More importantly, the recording stage can use common analog recording hardware such as standard CCD video cameras and analog VCRs, greatly reducing overall system cost. The quality of the video can be

controlled by the quality and recording format of the analog video, providing a range of solutions with variable cost. We use low cost equipment (consumer-grade SVHS VCRS), so the image quality is relatively low. One disadvantage of this approach is that the digital video capture process is longer, since each analog video must be digitized separately. For our purposes, this cost was tolerable. A more detailed report on the design and implementation of this architecture is also available [42].

B.2.1 Real-Time Multi-Camera Analog Video Recording

The video capture system must capture multiple video streams in real time. Standard VCRs record a single video channel in real time for long duration, so using one VCR per video camera ought to be an effective design strategy: the cost is low, the components are reliable, and the system would scale easily. The main obstacle to implementing this approach is that video from independent VCRs will not be synchronized, so there will be no way to relate video frames from one camera to those of another.

Our solution to this problem requires two steps. First, all the cameras are sent an electronic synchronization signal so that all the cameras open their shutters simultaneously, which will synchronize the video streams. Next, every field of each camera's video is time stamped with a common Vertical Interval Time Code (VITC) before being recorded onto video tape. The VITC allows the video to be re-synchronized after digitization. This time-code contains the hour (00-23), minute (00-59), second (00-59), frame (00-29), and field (0 or 1) for each video field.

B.2.2 Off-line Multi-Camera Video Digitization

At the conclusion of a multi-camera analog video recording session, the video must be digitized. The common VITC across all video cameras greatly simplifies this process. The operator selects a range of VITC codes to be digitized, and then runs an automatic digitization program, which digitizes only the desired video frames. The video digitizer itself does not need to support VITC timecode itself, since the software can read and interpret the VITC in real time.

B.3 Camera Placement

The placement of cameras around the workspace can affect overall system performance in a number of ways. Most importantly, if no camera sees a part of the scene, then the 3D digitization system will be unable to model that scene element. The placement also affects the quality of the geometric and appearance models.

B.3.1 Clustered vs. Uniform Camera Distributions

One aspect of camera placement is the distribution of cameras around the work space. One approach is to cluster cameras together, which can improve the correctness of stereo correspondence by reducing changes in visibility and appearance from camera to camera within a cluster. A uniform camera distribution will make the correspondence problem in stereo more difficult, but will reduce the visibility problem, i.e., the risk of not observing an important element of the scene. Since visibility is critical to modeling process, we chose a nearly uniform camera distribution.

B.3.2 Effects of Camera Placement on Image Resolution

Another aspect of camera placement is its impact on image resolution. As the camera is moved closer to the work space, the field of view must increase in order to capture the full space. For objects near the front of this space, the difference in resolution is small, but the resolution changes dramatically for objects near the back of the work space, as shown in Figure B.1. If we assume that the workspace is defined by a cube with one face parallel to the image plane, and that the optical axis of the camera intersects this face at its center, then the variation in resolution can be expressed as the square of the ratio of the depth of the near cube face to the far cube face:

$$K = \left(\frac{d}{d+w}\right)^2 \tag{B.1}$$

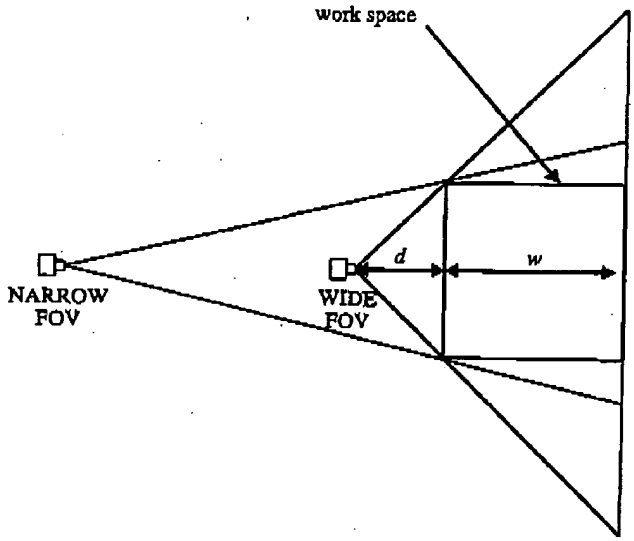

Figure B.1: Variation in image resolution across the work space. A camera close to the workspace must have a wide field of view (FOV) to see all of the near space. As a result, the resolution of objects near the back of the work space goes down rapidly compared to a camera with a longer distance d and narrower field of view.

where d is the depth to the near surface, w is the width of the work space, and K is the ratio of image pixels filled by the back side of workspace to pixels filled by the front side. Because of the constraints of the dome, our cameras had to be placed very close to the work space, with $d \sim 1$ meter, and $w \sim 2$ meters, yielding $K \sim 0.111$. (The camera labeled "WIDE FOV" actually approximates this configuration.) Thus, the far side of the work space has nearly an order magnitude lower resolution than the near side.

B.3.3 Density of Cameras

One of the most difficult factors to determine in designing a real multi-camera system is the number, or density, of cameras that are actually needed to observe the scene. Ideally, we want an optimal solution, not just a sufficient one. However, it is not obvious even how to define this sampling problem, let alone solve it. We therefore limit ourselves here to listing factors that must affect the solution. The most important factor is the 3D scene itself. Large areas of self-occlusion require a dense camera distribution to be able to see into the occlusion. Without some knowledge of the contents of the scene, little can be said about the camera density. Another major factor is image resolution, since it determines the

smallest resolvable (i.e., observable) 3D feature. As image resolution decreases, the smallest feature size increases. If we approximate scenes with occluded regions that are smaller than the smallest feature by similar scenes that close, or fill in, these occluded regions, then the image resolution bounds the complexity of the camera system necessary to completely observe the scene. The final major factor is the impact of camera spacing on the stereo algorithm. Changes in camera spacing can affect accuracy, precision, and coverage in stereo, since the correspondence search becomes increasingly difficult with wider camera separation. Finally, for scenes with minimal self-occlusion, the field of view of the cameras also may be important, since very few cameras are necessary to see the foreground structure. In this situation, the camera number may drop so low that portions of the background are not observed. If the system objective is only to observe the foreground, this factor can be ignored, and with reasonably complex scenes, this factor is negligible in comparison to the others. Although beyond the scope of this paper, this topic is worthy of a more detailed analysis.

In the design of 3D Dome, we predominantly focused on the needs of stereo while keeping system cost to reasonable levels, resulting in a camera spacing of approximately 60-70 cm. This baseline distance is relatively large for the 1-5 meter distance between each camera and the working volume of the dome – for example, the stereo machine [26] used a baseline that was an order of magnitude smaller for a similar range of depths – but our experiments have verified that stereo successfully operate on this narrow baseline. For a simple object such as a sphere, each surface point is observed by as many as all 51 cameras. With complex self-occlusion as in the examples shown throughout this thesis, at times only a few cameras see into highly complex regions.

B.4 Calibration

The video capture system provides only raw video for future processing. In order to make use of this data, the camera system must be calibrated. Each camera is modeled with the Tsai camera model, discussed in Appendix A. The calibration process fits the parameters of the general camera model to each real camera. This process requires a set of 3D points

in known positions along with the image projections of these points for each camera. In systems with only a few closely spaced cameras, this information can be collected by imaging a known 3D object in all cameras. Because of visibility constraints, however, this method does not extend easily to the calibration of a large volume surrounded by cameras.

We have developed two methods for dealing with this problem. The first approach is to decouple the calibration of intrinsic and extrinsic camera parameters. If the intrinsic parameters are known, then a single planar set of dots can be used to calibrate the extrinsic parameters. Since all cameras can see the floor of the dome, markers put directly on the floor in known positions can be used as the known 3D points on a plane for extrinsic parameter calibration. The second approach is to design and use a calibration object that avoids most of the visibility problems inherent in the multi-camera system. Then this object is moved through the volume to create a set of known 3D calibration points.

B.4.1 Decoupling of Intrinsic and Extrinsic Parameters

The decoupling approach involves two steps. First, to get the intrinsic parameters, each camera is removed from the dome and placed on a controlled imaging platform mounted in front of a small calibration plane. This plane has dots at known positions, and can be moved along the surface normal to create a set of 3D points. The plane is square, 40 cm on a side, and is moved 40 cm along its surface normal. By imaging the plane at several points along this motion, the projections of the 3D points can be determined. This data is fed into the calibration process to compute a full camera model. Second, the camera is then placed into its final position on the dome and used to image dots on the floor of the dome. These known 3D points and their image projections are then fed back into the calibration process, which now holds the intrinsic parameters fixed while optimizing the extrinsic camera parameters.

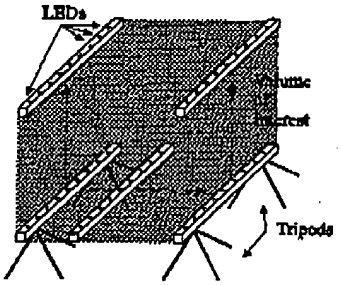

Figure B.2: The calibration bar assembly. The calibration bar contains 11 LEDs spaced 20 cm apart. The bar is mounted on two tripods, which provide known vertical positioning. Sliding the tripods orthogonally to the bar provides the remaining motion to define the 3D volume of calibration points.

B.4.2 Direct Calibration

In order to calibrate all cameras directly in their desired positions, the scene must be instrumented with a known 3D object. The object should span the 3D volume to obtain best results, but such an object is likely to occlude itself in many viewpoints. Our strategy for addressing this problem was to build a simple object with minimal self-occlusion and then build a mechanism to move this object through the volume to known 3D positions. This approach is philosophically identical to the method of calibrating the intrinsic parameters in the previous method, but the scale is much larger. The intrinsic method calibrated a volume of about 0.1 m^3, while this method must calibrate a volume of almost 10 m^3. Because of the physical dimensions involved, we chose to build only a bar rather than a full planar object. The bar also has less self-occlusion, since it must be viewed nearly end-on to obscure itself, while the plane could occlude itself from viewpoints near the plane.

The bar defines one horizontal dimension of the 3D volume. In order to define the vertical dimension, the bar is mounted on tripods whose vertical positions have been calibrated to raise the bar to known heights. The final dimension is defined by lateral translation of the bar-tripod assembly, with known positions marked on the floor. The bar has 11 LEDs mounted 20 cm apart. The floor is calibrated with markers 20 cm apart in both dimensions, as is the vertical motion of the tripods that the bar is mounted on. During calibration, this bar is swept through the volume along the lateral and vertical axes to generate known 3D points imaged in all cameras (see Figure B.2). Since we have three dimensional calibration data, we can now combine the extrinsic and intrinsic calibration steps into one.